Electronic Government and Electronic Participation
H.J. Scholl et al. (Eds.)

doi:10.3233/978-1-61499-670-5-99

Sustainable Linked Open Data Creation: An Experience Report

Eduard KLEIN [a,1], Stephan HALLER [a], Adrian GSCHWEND [a,b] and Milos JOVANOVIK [c,d]

[a] *E-Government Institute, University of Applied Sciences, Bern, Switzerland*
[b] *Zazuko GmbH, Biel, Switzerland*
[c] *Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, Macedonia*
[d] *OpenLink Software, United Kingdom*

Abstract: A flexible platform supporting the linked data life-cycle has been developed and applied in various use cases in the context of the large scale linked open data project *Fusepool P3*. Besides the description of the aims and achievements, experiences from publishing and reusing linked data in public sector and business are summarized. It is highlighted that without further help it is difficult for domain experts to estimate the time, effort and necessary skills when trying to transfer the platform to other use cases. Applying a new publishing methodology turned out to be useful in these cases.

Keywords: linked data, semantic enrichment, linked data life-cycle, data publishing, data integration, resource description framework, data management

1. Introduction

The exploitation of the Internet for intelligent knowledge management has been worked on for many years and it still remains one of the main challenges for the scientific community with added value for business, public bodies and civil society. In this attempt, the web is not only used in a classical way for publishing (unstructured) documents as HTML pages, offering online services like shopping, booking or text-based search engines, but also as a platform for processing and managing structured information. It appears in the form of data, which is published, interlinked and integrated with other structured information as linked data [1], that can subsequently be browsed or queried.

Annotated with appropriate vocabulary terms from ontologies, this interlinked structured information can not only be searched by keywords, but on a semantic level, thus laying the foundation for the Semantic Web [2]. Through linked data, information and services on the Internet and in web-based applications and mobile apps can and have already been enriched in a sophisticated way, although in the broad public it is not yet noticed as a big bang, since it comes in form of a quiet revolution [3]. Facebook's Knowledge Graph, Google's Hummingbird and Bing's Satori are examples of improv-

[1] Corresponding Author: eduard.klein@bfh.ch

ing services through semantic search technologies, revealing the revolution's silence through incrementally improving the services in small iterations while digesting constantly information from different sources.

In the e-Government domain, the use of linked open data (LOD) is spreading, as public authorities realize its benefits – not only regarding the transparency of governmental processes, but also as a driver for economic innovation: the availability of machine-readable semantically enriched open data enables SMEs and other entities to develop and provide new value-added services and applications. However, while public authorities in democratic countries around the globe have already or are developing a strategy for Open Government Data (OGD), only a fraction of those already take the additional step of provisioning the data as LOD through SPARQL endpoints. Take for example Switzerland: An e-Government strategy is in place both on the federal level (since 2007) as in most cantons; in addition, an OGD portal[2] as single point of entry for all OGD data in Switzerland has been established in February 2016. A service platform for LOD however is only available in a pilot stage with currently only a limited set of data.[3] One of the main roadblocks hindering a wider adoption of linked open data is that authorities shy away from the additional effort needed to convert OGD to LOD. This was also one of the key motivators to start the Fusepool P3 project.

Meanwhile, the Linked Data paradigm has fostered and propelled the emergence of numerous research projects and software products with focus on LOD [4]. Currently, the most prominent output of the LOD movement is visualized in the LOD cloud,[4] the core of which is formed by the data sets of DBpedia [5] and GeoNames.[5] Moreover, many domain-specific applications have evolved [6], often with an exploratory focus.

Inherent to LOD applications is the processing of data analogous to ETL processing in the data warehouse domain, but with more complex operations such as data extraction, enrichment, interlinking, fusing and maintenance. While these can be automated to a certain degree for a specific domain, a lot of manual work is still necessary, e.g., for mapping tasks. This data processing is part of the linked data life-cycle [7], that occurs with different complexity, among others depending on the data sources and the requirements of the target applications. In one way or another, the linked data life-cycle is integral in research projects like LOD2 [8], LATC [9], GeoKnow [10] and Fusepool [11].

In this paper we describe experiences from Fusepool P3 [12], a large scale EC-funded FP7 project with a focus on publishing and reusing linked data. The research goal was to develop enhanced products and services based on the exploitation of linked data in the context of the tourism domain. In the next section, the project goals are summarized, followed by a description of the architecture of the integrated data platform. Next, experiences from the project are pointed out, before concluding with aspects about the transfer of the research results to other application contexts.

[2] http://opendata.swiss/
[3] http://lindas-data.ch/
[4] http://lod-cloud.net/
[5] http://www.geonames.org/

Of the main findings we have learned that the Fusepool platform can significantly simplify the publishing of data as linked open data. Regional authorities in Trento and Tuscany were thus enabled to provide tourism-related data that form the basis for novel applications. Reflecting several completed use cases showed that additional advice and recommendations are essential for transferring project results to other use cases. A new publishing methodology, described below, allows for recording information on completed LOD projects and helps estimating and planning new LOD use cases.

2. The Fusepool P3 Platform

The main goal was to facilitate publishing and reuse of linked data in a more seamless way, based on a thriving data market economy with data providers, enhancers, and component developers along the linked data value chain [11]. In order to facilitate publishing and processing of Linked Data within a single platform, a set of loosely coupled, modular software components, compatible with the Linked Data Platform (LDP) best practices [13], has been developed. These software components work closely together, supporting the multilingual data value chain, to achieve the following tasks: *revealing* data from structured and unstructured sources, *refining* data through text extraction and enrichment, and *running* the linked data ecosystem through data-driven applications (Fig. 1).

Supported by appropriate backend tools described below, and a high degree of automation in data processing, the Fusepool platform has successfully been deployed in several research projects, including the preservation of intellectual property of SMEs in the patent domain and in tourism use cases.[6]

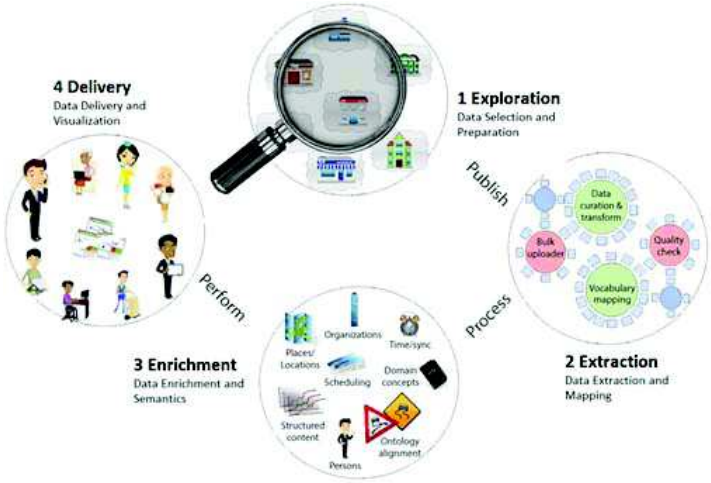

Figure 1: Elements of the *Fusepool P3* data value chain; *Fusepool* derives its name from the idea of fusing and pooling linked data with analytical processing on top of it, and *P3* abbreviates *Linked Data Publish-Process-Perform*

[6] http://fusepool.eu/

2.1 Architecture

We aim at providing a single platform for the linked data life-cycle. To achieve this, the Fusepool platform architecture is based on loosely coupled components communicating via HTTP and exposing RESTful APIs exchanging RDF [14]. This leads to reusability of components, enables distributed development and makes it easier for developers to understand and extend the software, thus ensuring its longevity.

RESTful RDF is the platform's native interaction method, meaning that there are no proprietary data access APIs in place. Platform components, as well as third party applications, communicate using generic RDF APIs. In Fig. 2, the Fusepool platform architecture is depicted.

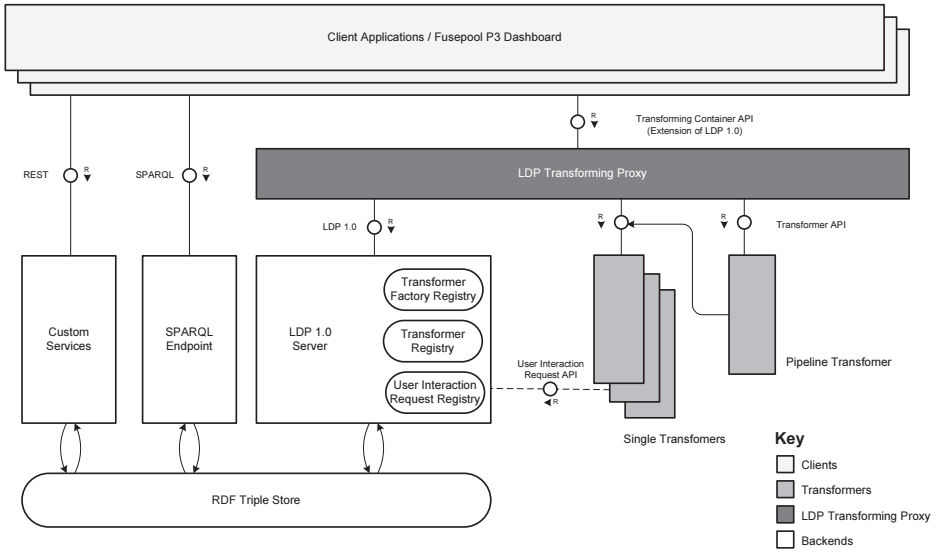

Figure 2: The P3 platform in Fundamental Modeling Concepts (FMC) Notation

The diagram shows how the Fusepool P3 dashboard – the main user interface to interact with the platform – and other clients access the Fusepool platform primarily via an LDP Transforming Proxy, an extension of the LDP 1.0 specification which uses the REST-based Transforming Container API to enable RDF data generation and annotation from input data. The proxy transparently handles transformation processes by calling the actual transformers in the background, and once the process has finished, it sends back the data to the LDP Server. The clients can also directly access transformers via their REST API (the *Transformer API*) or use a SPARQL 1.1 endpoint.

As a result, the architecture does not require a common runtime for its components. Every component, including all transformers, is by default run as an individual process acting via HTTP as the interaction interface. The exception to this are the backend related components (LDP, SPARQL, the RDF Triple Store and possible custom backend services) which may be more tightly coupled, i.e., they may be run in the same

runtime environment, due to non-functional requirements such as performance or other resource cost.

2.2 Components

The P3 platform is composed of three core components: transformers, the LDP Transforming Proxy and backends. Applications such as the Fusepool dashboard are external components which mainly use standard interfaces such as LDP or SPARQL. The platform components communicate with each other via REST over HTTP. RDF is used as the data model and exchange format in all communications, with the exception of the use of SPARQL. All standard RDF serializations may be used, with Turtle being explicitly supported by all components implemented to date. Besides LDP and SPARQL, the interaction between the components, as well as with external clients, is regulated by APIs and the Fusepool Annotation Model (FAM) which are briefly explained below.

Transformers. Data transformation components are responsible for transforming data from legacy formats (structured and unstructured) into RDF, and adding or refining annotations to input data. In the Fusepool platform there are two families of transformers: *RDFizers* and *Annotators*. The former transform non-RDF data to RDF, and the latter enrich data in any format with RDF annotations.

Transformers are identified by a URI, which is the entry point for the RESTful Transformer API defining the interaction with the transformer components. This API supports both synchronous and asynchronous transformers. While a synchronous transformer returns the transformation result right away in the response to the transformation request, an asynchronous transformer delivers its result at a later time. Asynchronous transformers may also require some user interaction in order to deliver their results.

A *pipeline transformer* invokes a list of transformers in sequence, passing the output of a transformer as input to the next transformer. This enables chaining of multiple transformers to perform more complex tasks.

The above-mentioned annotators are expected to produce annotation from textual content, either unstructured or extracted from any other structured format. All annotators produce RDF using FAM [14]. This is an important approach for piping annotators and allowing configurations using multiple annotation services. The base structure of FAM is fully compatible with Open Annotation [15], but defines some additional relations to ease the consumption of annotator results – especially the retrieval of selectors for annotations.

LDP Transforming Proxy. This is an HTTP Proxy that is used as a reverse proxy in front of an LDP Server. It intercepts POST requests against LDP Containers (LDPCs) which are marked as *Transforming Containers* and then it (a) forwards the request to the proxied LDP instance, and (b) sends the contents to the transformer associated with the container. Once the result of the transformation is available, the LDP Transforming Proxy will post it to the LDPC as well. In this way, the Transforming LDPC holds both the original and the transformed data. A transforming LDPC can have a pipeline trans-

former associated with it, should multiple transformers be executed over the POSTed data.

The *Transforming Container API* is defined as an extension to the LDP specification to allow special containers to execute a transformer when a member resource is added via a POST request. This allows documents to be automatically transformed when they are added to a LDPC, and having both the original data and the transformed data as a resource inside the Transforming LDPC. This process is supported via the LDP Transforming Proxy.

The *User Interaction Request API* describes how an LDPC is used to maintain a registry of requests for interaction. Its purpose is to provide support for components which require user interaction during their lifetime, such as transformers requesting a user input. According to the API, components submit a URI to the mentioned registry, and remove the URI once the interaction is completed. A UI component can then provide the user with a link to the submitted URI. The component is free to present any web-application at the denoted URI suitable for performing the required interaction.

Backends. The platform can use both Apache Marmotta and Virtuoso Universal Server as backends, which provide the generic LDP and SPARQL interfaces and data persistence in an RDF Triple Store. However, based on the architectural approach, any other tool which supports the LDP and/or the SPARQL standards can be used as the platform backend as well.

3. Experiences

Our experiences with the Fusepool platform are best explained by the example of our two initial stakeholders in the Fusepool P3 project, namely two touristic regions in Italy: Provincia Autonoma di Trento (PAT) and Regione Toscana (RET). They have been publishing open data and are supporting the development of applications and services in the tourism domain for some time. During this time both partners gained valuable experience in data creation, maintenance and publication.

3.1. Limitations in Publishing Open Data

PAT and RET first started publishing data sets which were considered strategic. In Italy in general but also in the two regions Tuscany and Trentino, one of the most important businesses is tourism. This also includes linked and related industrial activities around tourism. Thus the regions are struggling with one particular question: How can they support and push tourism by changing their daily operations.

Both partners provide a CKAN based open data portal,[7] which aims at data publishers providing tools to find and use data. The data quality depends on the data provider. Except adding some meta information, the data that gets pushed into the system is the data which is made available to the user.

[7] http://ckan.org/

At project start, open data from PAT and RET was only available in particular data formats like CSV, KML, XML and JSON. App developers had to download the raw data and process it using their own ETL processes. With every update of the raw data, this process had to be triggered manually for every single application using this data. If the format of the raw data changed, the process had to be adjusted and could not be automated. With every new data source, maintenance complexity of these open data sets and its apps increased.

3.2. Linked Data Life-Cycle

Reducing the complexity for consuming open data requires that the necessary ETL work is done up-front, ideally by the data owner or someone with domain knowledge. Furthermore, the data should preferably be published as a service and without the need for running separate database servers and other services. This is where linked data and its RDF technology stack come into play. With its open, non-proprietary data model using W3C standards such as SPARQL and HTTP, RDF is used as Lingua Franca using well-known schemas and ontologies.

In the classic document-centric web not much is known about the relationship between two pages as links between them are untyped. RDF links far more granular entities than pages, i.e. single attributes of an object, and defines relations between data items in schemas and ontologies. Best practices recommend publishing these schemas and ontologies also as RDF, thus making them publicly available in a machine-readable form.

3.3. Applying the Linked Data Life-Cycle

Experiences with applying the linked data life-cycle using the Fusepool platform were made in preparation for and during a hackathon at the Spaghetti Open Data Event,[8] where the initial versions of two linked open data applications based on data from the Province of Trento were developed.

In the first one, a web application called “LOD events eXplorer” allows events in the Trento region to be browsed, and information and pictures of nearby points of interests (POIs) are also shown (see Fig. 3). The developers could easily transform the original data set provided as an XML feed into RDF using the XSLT transformer provided by the Fusepool platform and store the results in the data store of the platform, making it accessible through SPARQL queries.

The most time-consuming manual task in doing so was to develop the XSLT file that defines the mapping from the XML elements to the appropriate RDF model; creation of the mapping required developer skills and was a matter of a few hours, including familiarization with the tool and environmental setting. The subsequent transformation of the data itself however took place in a matter of seconds only. RDFizing and interlinking other data such as nearby POIs and images from DBpedia turned out to be an easy and less complex task compared to the development of the initial mapping.

[8] http://www.spaghettiopendata.org/

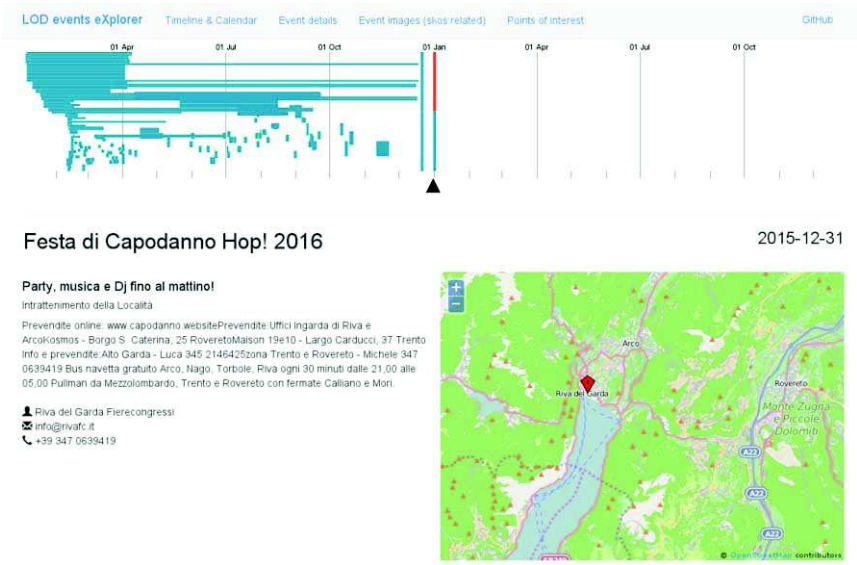

Figure 3: The LOD events eXplorer application, showing events in the Trento region

A second application enables tourists to follow the footsteps of historical figures from Trentino. They can read about these people, see where they lived and find POIs and restaurants nearby. This mobile application – available in the respective app stores for iOS and Android under the name "In The Footsteps: Trentino" – is based on several open data sets available on the CKAN site operated by the region of Trento: namely, historical characters, restaurants, architectural and artistic heritage plus POIs. These were transformed in a similar fashion to linked open data on the Fusepool platform. For additional information, data from Wikipedia and Yelp was also linked in. The development time and the necessary skills turned out to be comparable to the LOD events eXplorer application.

3.4. A Linked Data Publishing Methodology

Reflecting several LOD use cases, including those from the previous section, a common methodology could be distilled, comprising of analysis, design and implementation steps. It turned out to be very helpful to externalize the findings in a LInked DAta PUblishing MEthodology (LIDAPUME), consisting of a methodology schema (Fig. 4) and a template for orientation and guidance (Table 1).

Compared to other publishing methodologies, such as those used in LOD2 [8] or LATC [9], non-technical steps are also under consideration here, as opposed to the solely technical data life-cycle steps which are often used in related approaches. This more holistic approach promotes the documentation of essential tasks which proved to be helpful answering questions like "How long will it take to develop a use case with this platform?" or "How many technical skills are necessary in order to achieve this?". The LIDAPUME steps are described in more detail in [16].

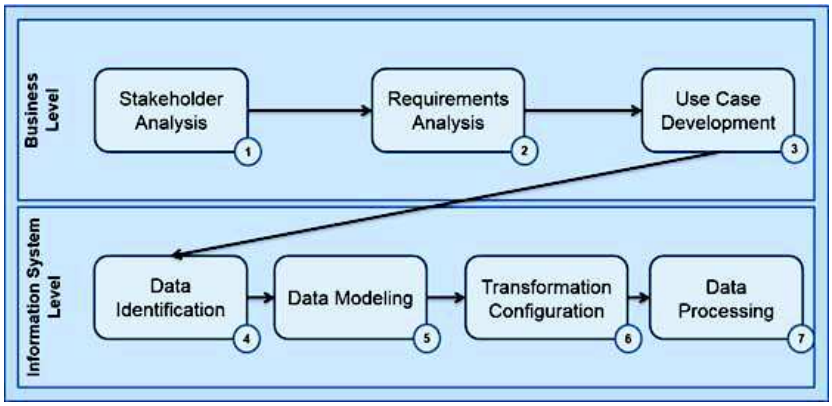

Figure 4: LIDAPUME schema, a linked data publishing methodology (from [16])

The template in Table 1 shows an instance of LIDAPUME, allowing annotations of essential use case aspects. The template has been completed in the context of the *Swiss Archive Use Case*, described below.

Step	Activities	Skills	Effort
Stakeholder Analysis	Identifying Swiss cantonal archives that want to participate in the user case. Partially given as it was initiated by some of the archives itself (5 stakeholders).	Analytical	1D
Requirements Analysis	Interviews with each archive. Identifying a topic available in all archives.	Analytical	1D
Use Case Development	Developing ideas of an initial user interface for the use case.	Analytical, prototyping	1D
Data Identification	In-place work with each stakeholder, data source analysis and data export options.	Analytical, technical	2D
Data Modelling	Identifying appropriate vocabularies. Mapping data from the data source with OpenRefine or XSLT configurations. Partially in-place at stakeholder.	Domain knowledge, modelling, technical	5D
Transformation Configuration	Enhancing data with named entity recognition and interlinking with reference datasets such as GND[2].	Modelling, technical	3D
Data Processing	Execution of transformation, indexing of data. Making it available on the final SPARQL endpoint.	Modelling, technical	3D

Table 1: LIDAPUME template for the Swiss Archive Use Case (1D=1 effort-day)

Using the methodology and the template turned out to be a good starting point for LOD use case planning, with regards to completeness of the planning, necessary project skills and project duration. Having experience from completed projects at hand, allows for better estimation and shortens the learning curve.

The LIDAPUME methodology and template have been validated for several use cases which are described in more detail in [16]. Besides the above described use cases, it has been applied in enhancing the FU Berlin library content through an LOD use case, called *Library Keyword Clustering*, and in the *Swiss Archive Use Case* [17].

4. Conclusion and Outlook

In the past, a lot of time and energy was invested in providing tools for converting particular sets of data to linked data. Several FP7 projects such as LOD2 [8], LATC [9] and GeoKnow [10] funded transformation of large linked data datasets which are now available within the linked open data cloud. The Fusepool platform provides additional value in the domain as it brings an integrated set of components that allow open data from various sources to be easily published as linked open data, enabling development of useful applications, like the examples described in this paper. The tools provided are not domain-specific. While the current use cases have mainly been in the tourism domain, the methods can be applied equally well to other domains; we recently used the platform to successfully transform around five million public records from the Swiss Federal Archive[9] and four Swiss cantons and interlink it with GND, a universal authority file.[10]

The most time-consuming task in order to promote data to the 5-star level [18] is in defining the mappings of the original data sets to a linked data model. This requires domain knowledge and close cooperation with domain experts. Once that one-time effort has been done, the actual transformation of data can be automated such that new data sets of the same type are to be published, they are transformed to linked data and added to the RDF triple store.

To address this one-time effort, it turned out that two basic questions have to be answered, namely "What are stable identifiers in the particular dataset?", and "What is the meaning of the data and how does it map to existing schemas and vocabularies?"

Answering the first question will help to coin stable URIs while the second question will make data more useful for new data publishers. Integrating services like Linked Open Vocabularies (LOV) [19] in P3 transformers support domain specialists in mapping data to commonly used vocabularies. It is commonly recommended that the focus should be on reusing existing vocabularies where possible and repurposing and extending them where necessary only.

Experience has shown that the tools and technologies of the Fusepool platform for publishing and reusing linked data are well suited for data publishers with technical skills. For users with fewer technical skills additional help is necessary, whether it is in the form of advice from developers or – preferably – in the form of guidelines and more intuitive wizard-style tool guidance. Even for developers the learning curve is not insignificant, in our use cases several iteration steps were necessary in order to become familiar with the tooling environment and the data life-cycle processes.

To make sure these datasets and tools are maintainable, it is important to empower data owners to run these processes on their own. Fusepool P3 provides some of the necessary glue to integrate standalone components that were developed in the past and

[9] http://www.bar.admin.ch/
[10] http://www.dnb.de/DE/Standardisierung/GND/gnd_node.html

will be developed in the future. By providing *docker images*,[11] the Fusepool platform can be deployed within an organization within a few hours.

To have a sustainable linked data ecosystem, still more work is necessary on the user interface level. In a follow-up project, it is thus planned to work with data publishers to simplify the dashboard UI and to add a wizard-style tool guidance: For example, when the user selects an XML-based data set in a CKAN site that he wants to publish as linked data, the wizard will suggest using the XSLT transformer. The user still has the option to choose another transformer like BatchRefine (which adds batch processing capabilities to OpenRefine), but the wizard limits the possible user selections only to transformers that can take an XML file as input.

In addition, it is planned to develop a cookbook that gives non-technical users step-by-step instructions including screen casts on how to use the platform. It will be based on three typical user scenarios, considering first data and subsequently technical components:

1. Based on a concrete data set in a CKAN site. The cookbook explains the steps and the usage of additional tools that may be needed, e.g., how to create an OpenRefine configuration in order to publish data from a CSV-based format.
2. Based on a concrete data file. This is very similar to the first scenario, the difference being that the file is not retrieved from a CKAN site but available on a local drive.
3. Based on a known data structure and some sample data.

These changes and additions will hopefully simplify and improve the platform, allowing data publishers to use it without further help, hence significantly simplifying the task of publishing data as linked open data.

Acknowledgement: The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 609696.

References

[1] T. Heath and C. Bizer, "Linked Data: Evolving the Web into a Global Data Space," *Synthesis Lectures on the Semantic Web: Theory and Technology*, vol. 1, no. 1. pp. 1–136, 2011.
[2] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," *Sci. Am.*, vol. 284, no. 5, pp. 35–43, 2001.
[3] W. Hall, "Linked Data: The Quiet Revolution," *ERCIM News*, vol. 96, p. 4, 2014.
[4] F. Bauer and M. Kaltenböck, *Linked Open Data: The Essentials*. edition mono, Vienna, Austria, 2012.
[5] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann, "DBpedia - A crystallization point for the Web of Data," *J. Web Semant.*, vol. 7, no. 3, pp. 154–165, 2009.
[6] ERCIM, "ERCIM News," *News*, 2014. [Online]. Available: http://ercim-news.ercim.eu/en96. [Accessed: 22-Oct-2015].
[7] S. Auer, J. Lehmann, A. C. Ngonga Ngomo, and A. Zaveri, "Introduction to linked data and its lifecycle on the web," *LNAI*, vol. 8067, pp. 1–90, 2013.

[11] http://docker.com/

[8] S. Auer, L. Bühmann, C. Dirschl, O. Erling, M. Hausenblas, R. Isele, J. Lehmann, M. Martin, P. N. Mendes, B. Van Nuffelen, C. Stadler, S. Tramp, and H. Williams, "Managing the Life-Cycle of Linked Data with the LOD2 Stack," in *The Semantic Web–ISWC 2012*, 2012, pp. 1–16.
[9] LATC, "LATC," *LOD Around-the-Clock*, 2012. [Online]. Available: http://sourceforge.net/projects/latc. [Accessed: 22-Oct-2015].
[10] S. Athanasiou, D. Hladky, G. Giannopoulos, G.-R. Alejandra, and J. Lehmann, "GeoKnow: Making the Web an Exploratory Place for Geospatial Knowledge," *ERCIM News*, vol. 96, pp. 12–13, 2014.
[11] M. Kaschesky and L. Selmi, "Fusepool R5 linked data framework," *Proc. 14th Annu. Int. Conf. Digit. Gov. Res. - dg.o '13*, p. 156, 2013.
[12] A. Gschwend, A. C. Neuroni, T. Gehrig, and M. Combetto, "Publication and Reuse of Linked Data: The Fusepool Publish-Process-Perform Platform for Linked Data," in *Electronic Government and Electronic Participation: Joint Proceedings of Ongoing Research of IFIP EGOV and IFIP ePart 2015*, vol. 22, pp. 116–123.
[13] S. Speicher, J. Arwe, and A. Malhotra, "Linked Data Platform," *W3C Recommendation*, 2015. [Online]. Available: http://www.w3.org/TR/2015/REC-ldp-20150226/.
[14] R. Gmür, S. Fernández, J. Frank, R. Westenthaler, C. Blakeley, I. Kingsley, and A. Gschwend, "The Fusepool P3 Platform," 2014. [Online]. Available: https://fusepool.gitbooks.io/the_fusepool_p3_platform/content/d51-deliverable.html. [Accessed: 22-Oct-2015].
[15] R. Sanderson, P. Ciccarese, and H. Van de Sompel, "Designing the W3C open annotation data model," *Proc. 5th Annu. ACM Web Sci. Conf. - WebSci '13*, pp. 366–375, 2013.
[16] E. Klein, A. Gschwend, and A. C. Neuroni, "Towards a Linked Data Publishing Methodology," in *CeDEM 2016 (Conf. for e-Democracy and Open Governemt)*.
[17] J.-L. Cochard, A. Dubois, A. D. Gonzenbach, A. Gschwend, K. Lambert, S. Kwasnitza, M. Luggen, U. Meyer, F. Noyer, and T. Wildi, "Archival-Linked Open Data: practical and technical approach - A swiss collaborative project," in *Archives: Evidence, Security & Civil Rights (3rd ICA Conf.)*, 2015.
[18] T. Berners-Lee, "Linked data-design issues," 2009. [Online]. Available: http://www.w3.org/DesignIssues/LinkedData.html. [Accessed: 22-Oct-2015].
[19] P.-Y. Vandenbussche and B. Vatant, "Linked Open Vocabularies," *ERCIM News*, vol. 96, p. 21, 2014.